

Applied A.I. Solutions

Data Visualization Techniques

Professor
Daniel Vitaver, EMBA

daniel.vitaver@georgebrown.ca

DATA VISUALIZATION TECHNIQUES

Python libraries for data visualization

Python libraries

Data visualization in python is perhaps one of the most utilized features for data science with python in today's day and age.

The libraries in python come with lots of different features that enable users to make highly customized, elegant, and interactive plots.

- Matplotlib
- Seaborn
- Bokeh
- Plotly
- Others

Python libraries

- **Matplotlib**

- Matplotlib is a visualization library in Python for **2D plots of arrays**
- Matplotlib is written in Python and makes use of the **NumPy library**
- It can be used in Python and IPython shells, Jupyter notebook, and web application servers.
- It comes with a wide variety of plots like **line, bar, scatter, histogram**, etc. which can into understanding **trends, patterns, correlations**

Python libraries

- **Seaborn**

- Seaborn is a dataset-oriented library for making statistical representations in Python
- It is developed atop **matplotlib**, to create different visualizations
- It is integrated with **Pandas data structures**
- The library internally performs the required mapping and aggregation to create informative visuals

Python libraries

- **Bokeh**

- Bokeh is an interactive visualization library for modern web browsers
- It is suitable for large or streaming data assets
- It can be used to develop interactive plots and dashboards
- It works closely with **PyData** tools
- The visuals can be made **interactive** to serve a **what-if scenario model**

Python libraries

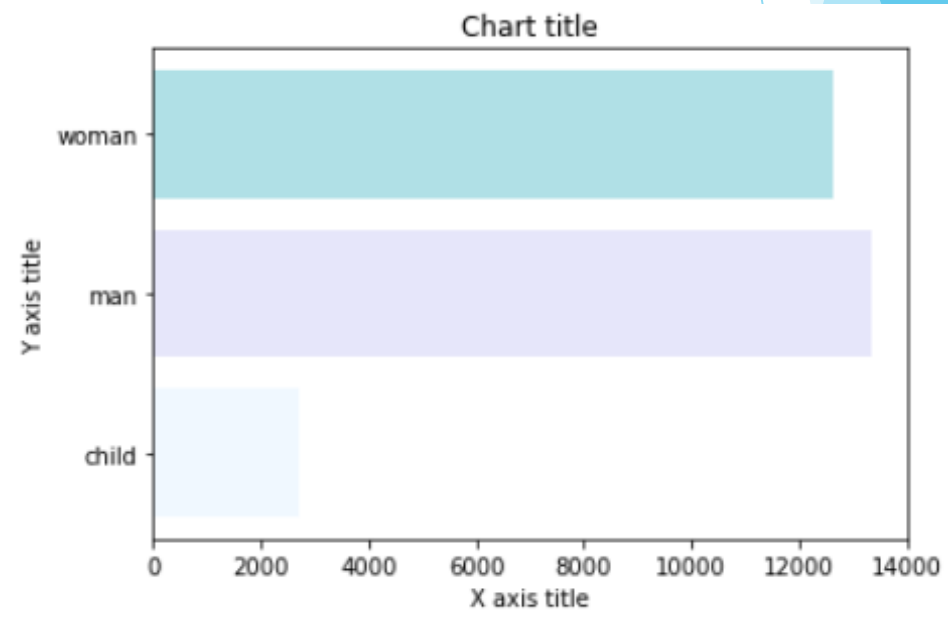
- **Plotly**

- plotly.py is an **interactive**, open-source, high-level, declarative, and browser-based visualization library for Python
- It holds an array of useful visualization which includes **scientific charts, 3D graphs, statistical charts, financial charts** among others
- Plotly graphs can be viewed in Jupyter notebooks, standalone HTML files, or hosted online
- Plotly library provides options for interaction and editing
- The API works perfectly in both local and web browser mode

Python libraries - Demos

• Matplotlib - Bar chart

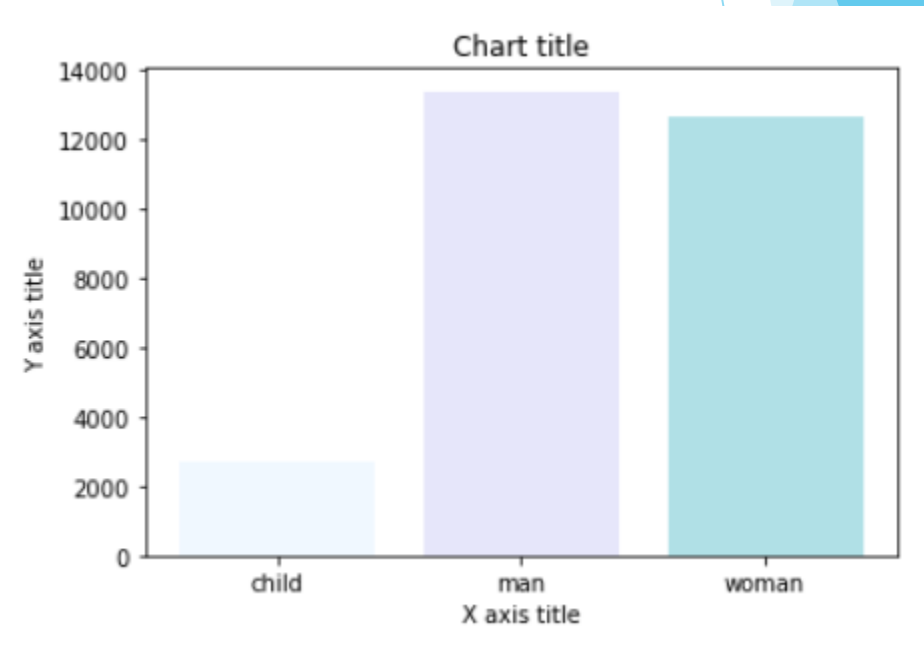
- #Creating the dataset
- `df = sns.load_dataset('titanic')`
- `df=df.groupby('who')['fare'].sum().to_frame().reset_index()`
- #Creating the bar chart
- `plt.barh(df['who'],df['fare'],color = ['#F0F8FF','#E6E6FA','#B0E`
- #Adding the aesthetics
- `plt.title('Chart title')`
- `plt.xlabel('X axis title')`
- `plt.ylabel('Y axis title')`
- #Show the plot
- `plt.show()`
- `x=""`



Python libraries - Demos

• Matplotlib – Column Chart

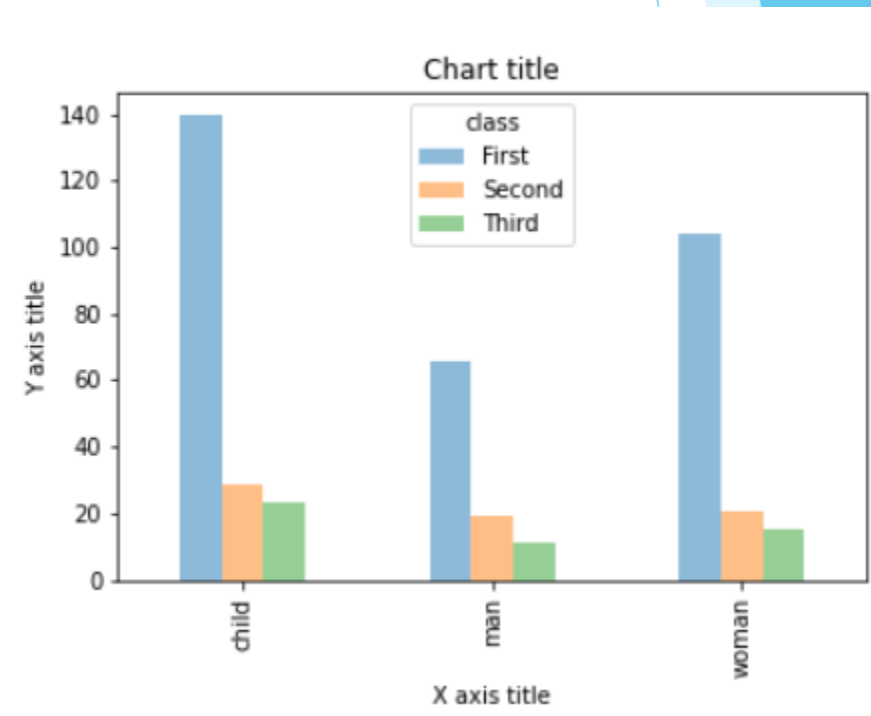
- #Creating the dataset
- `df = sns.load_dataset('titanic')`
- `df=df.groupby('who')['fare'].sum().to_frame().reset_index()`
- #Creating the column plot
- `plt.bar(df['who'],df['fare'],color = ['#F0F8FF','#E6E6FA','#B0E0E6'])`
- #Adding the aesthetics
- `plt.title('Chart title')`
- `plt.xlabel('X axis title')`
- `plt.ylabel('Y axis title')`
- #Show the plot
- `plt.show()`
- `x=""`



Python libraries - Demos

• Matplotlib – Grouped Chart

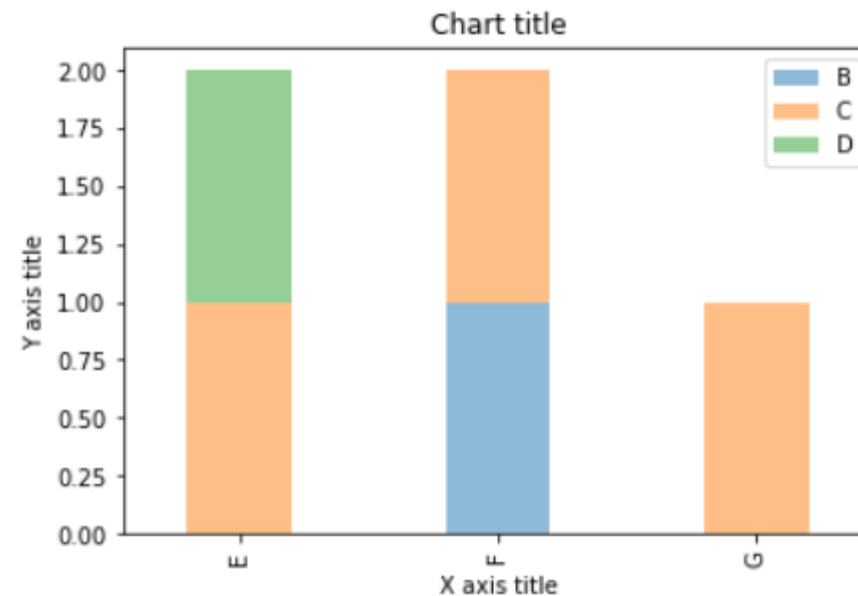
- #Creating the dataset
- import pandas as pd
- import numpy as np
- import matplotlib.pyplot as plt
- import seaborn as sns
- df = sns.load_dataset('titanic')
- df_pivot = pd.pivot_table(df, values="fare", index="who", columns="class", aggfunc=np.mean)
- #Creating a grouped bar chart
- ax = df_pivot.plot(kind="bar", alpha=0.5)
- #Adding the aesthetics
- plt.title('Chart title')
- plt.xlabel('X axis title')
- plt.ylabel('Y axis title')
- # Show the plot
- plt.show()
- x=""



Python libraries - Demos

• Matplotlib – Stacked Bar Chart

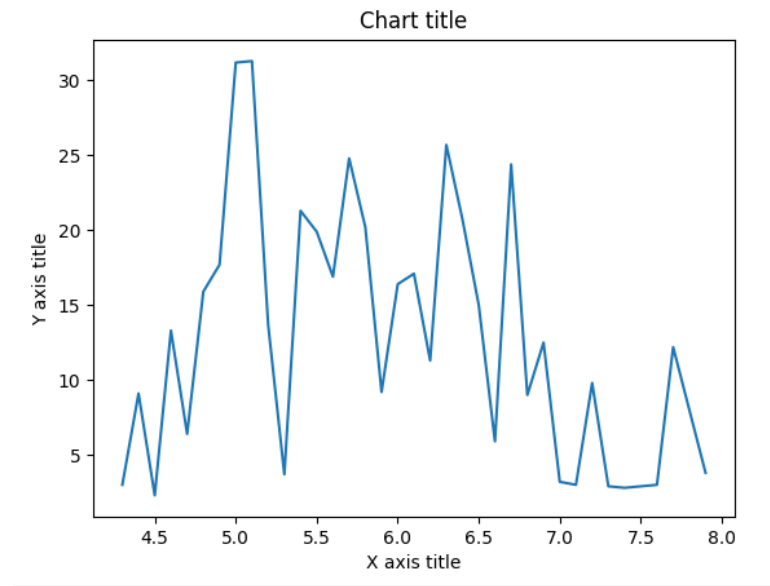
- **#Creating the dataset**
- `df = pd.DataFrame(columns=["A", "B", "C", "D"], data=[["E", 0, 1, 1], ["F", 1, 1, 0], ["G", 0, 1, 0]])`
- `df.plot.bar(x='A', y=["B", "C", "D"], stacked=True, width = 0.4, alpha=0.5)`
- **#Adding the aesthetics**
- `plt.title('Chart title')`
- `plt.xlabel('X axis title')`
- `plt.ylabel('Y axis title')`
- **#Show the plot**
- `plt.show()`
- `x=""`



Python libraries - Demos

- **Matplotlib – Line Chart**

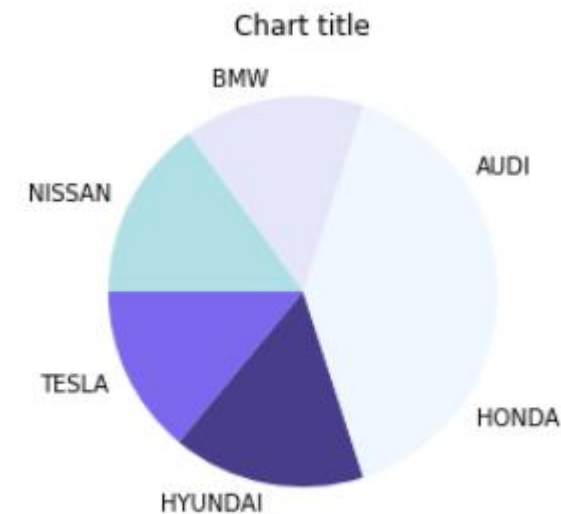
- **#Creating the dataset**
- `df = sns.load_dataset("iris")`
- `df=df.groupby('sepal_length')['sepal_width'].sum().to_frame().reset_index()`
- **#Creating the line chart**
- `plt.plot(df['sepal_length'], df['sepal_width'])`
- **#Adding the aesthetics**
- `plt.title('Chart title')`
- `plt.xlabel('X axis title')`
- `plt.ylabel('Y axis title')`
- **#Show the plot**
- `plt.show()`
- `x=""`



Python libraries - Demos

• Matplotlib – Pie Chart

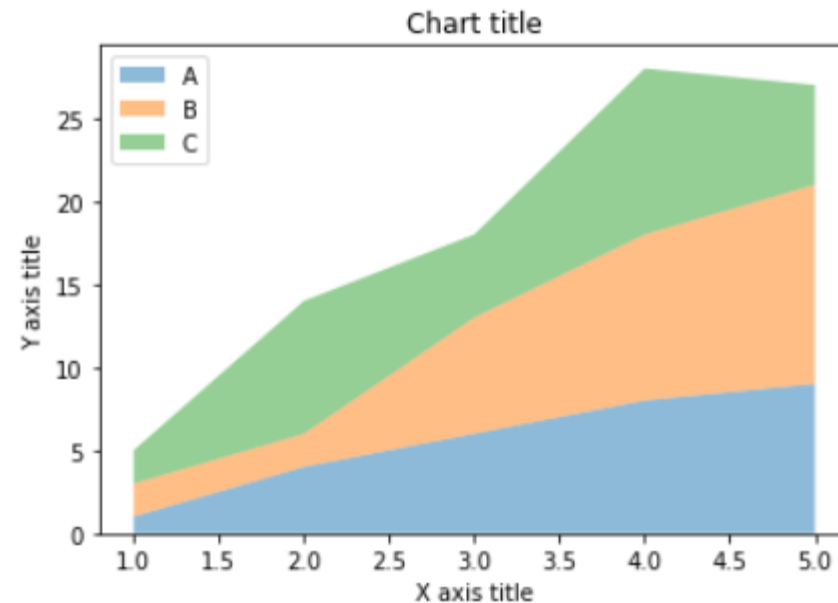
- `#Creating the dataset`
- `cars = ['AUDI', 'BMW', 'NISSAN', 'TESLA', 'HYUNDAI', 'HONDA']`
- `data = [20, 15, 15, 14, 16, 20]`
- `#Creating the pie chart`
- `plt.pie(data, labels = cars, colors = ['#F0F8FF', '#E6E6FA', '#B0E0E6', '#7B68EE', '#483D8B'])`
- `#Adding the aesthetics`
- `plt.title('Chart title')`
- `#Show the plot`
- `plt.show()`
- `x=""`



Python libraries - Demos

- **Matplotlib – Area Chart**

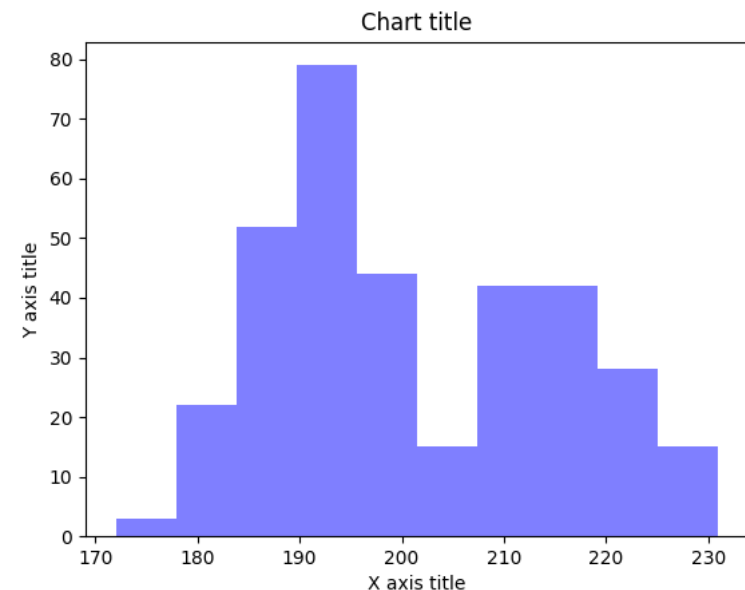
- `#Reading the dataset`
- `x=range(1,6)`
- `y=[[1,4,6,8,9], [2,2,7,10,12], [2,8,5,10,6]]`
- `#Creating the area chart`
- `ax = plt.gca()`
- `ax.stackplot(x, y, labels=['A','B','C'],alpha=0.5)`
- `#Adding the aesthetics`
- `plt.legend(loc='upper left')`
- `plt.title('Chart title')`
- `plt.xlabel('X axis title')`
- `plt.ylabel('Y axis title')`
- `#Show the plot`
- `plt.show()`
- `x=""`



Python libraries - Demos

- **Matplotlib – Column Histogram**

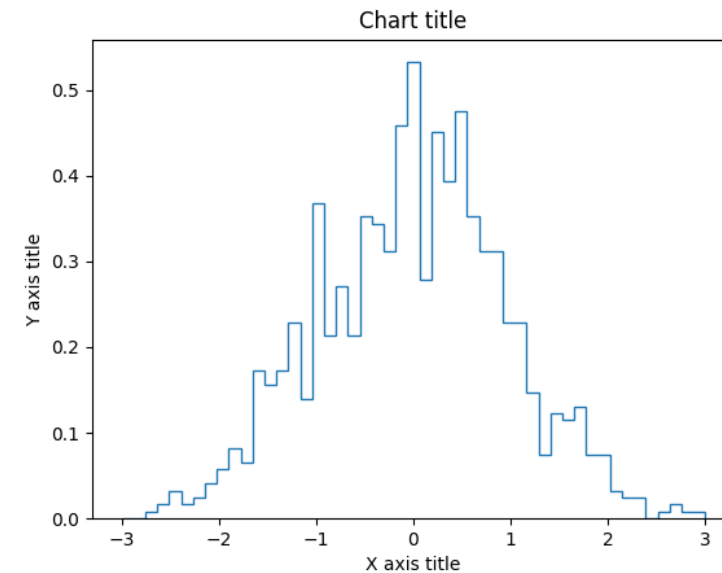
- `#Creating the dataset`
- `penguins = sns.load_dataset("penguins")`
- `#Creating the column histogram`
- `ax = plt.gca()`
- `ax.hist(penguins['flipper_length_mm'], color='blue', alpha=0.5, bins=10)`
- `#Adding the aesthetics`
- `plt.title('Chart title')`
- `plt.xlabel('X axis title')`
- `plt.ylabel('Y axis title')`
- `#Show the plot`
- `plt.show()`
- `x=""`



Python libraries - Demos

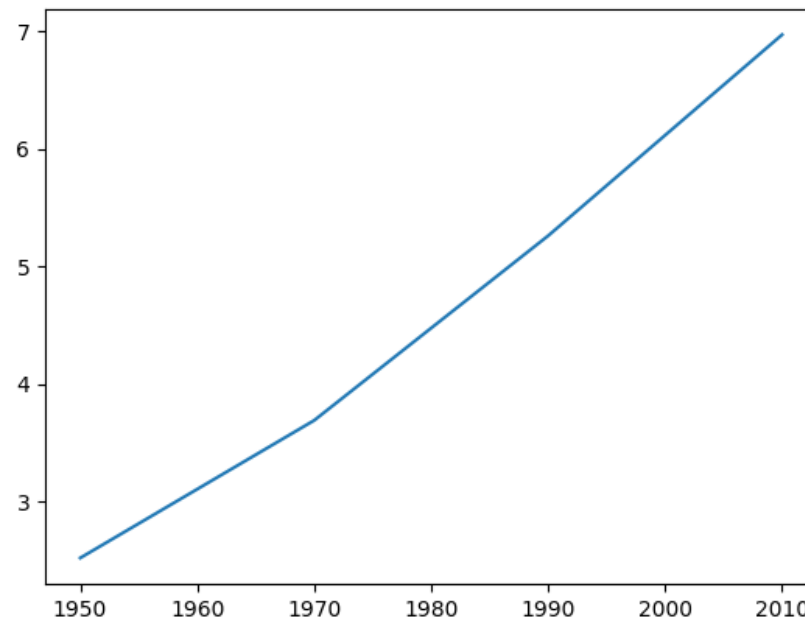
- **Matplotlib – Line Histogram**

- **#Creating the dataset**
- `df_1 = np.random.normal(0, 1, (1000,))`
- `density = stats.gaussian_kde(df_1)`
- **#Creating the line histogram**
- `n, x, _ = plt.hist(df_1, bins=np.linspace(-3, 3, 50), histtype=u'step', density=True)`
- `plt.plot(x, density(x))`
- **#Adding the aesthetics**
- `plt.title('Chart title')`
- `plt.xlabel('X axis title')`
- `plt.ylabel('Y axis title')`
- **#Show the plot**
- `plt.show()`
- `x=""`



Matplotlib: Plot a Function $y = f(x) = A x + B$

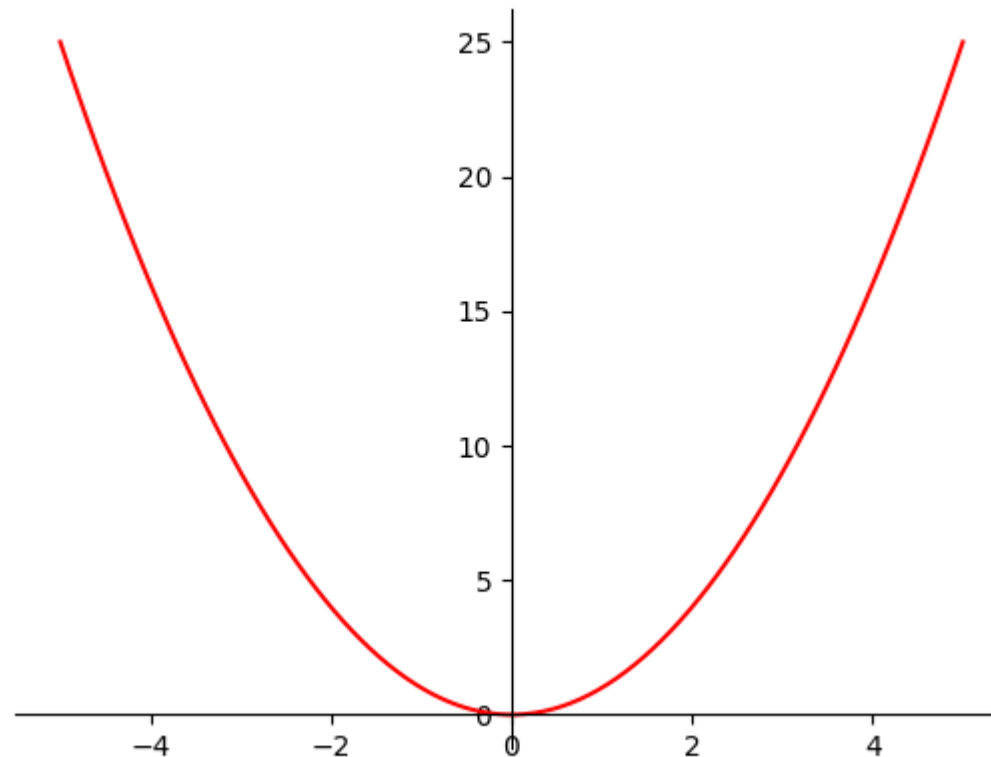
- `import matplotlib.pyplot as plt`
- `Import numpy as np`
- `#create dataset`
- `year = [1950, 1970, 1990, 2010]`
- `population = [2.52, 3.69, 5.26, 6.97]`
- `#create Line Chart`
- `plt.plot (year, population)`
- `#Show plot`
- `plt.show()`



<https://scriptverse.academy/tutorials/python-matplotlib-plot-function.html>

Matplotlib: Plot a Function $y = f(x) = Ax^2 + Bx + C$

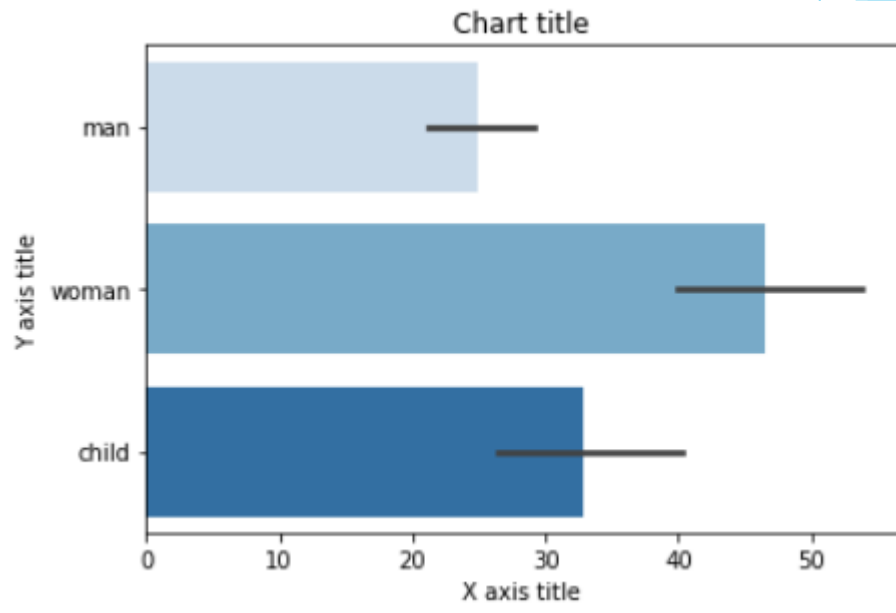
- import matplotlib.pyplot as plt
- import numpy as np
- # 100 linearly spaced numbers
- $x = np.linspace(-5, 5, 100)$
- # the function, which is $y = x^2$ here
- $y = x^{**2}$
- # setting the axes at the centre
- `fig = plt.figure()`
- `ax = fig.add_subplot(1, 1, 1)`
- `ax.spines['left'].set_position('center')`
- `ax.spines['bottom'].set_position('zero')`
- `ax.spines['right'].set_color('none')`
- `ax.spines['top'].set_color('none')`
- `ax.xaxis.set_ticks_position('bottom')`
- `ax.yaxis.set_ticks_position('left')`
- # plot the function
- `plt.plot(x, y, 'r')`
- # show the plot
- `plt.show()`



Python libraries - Demos

• Seaborn - Bar chart

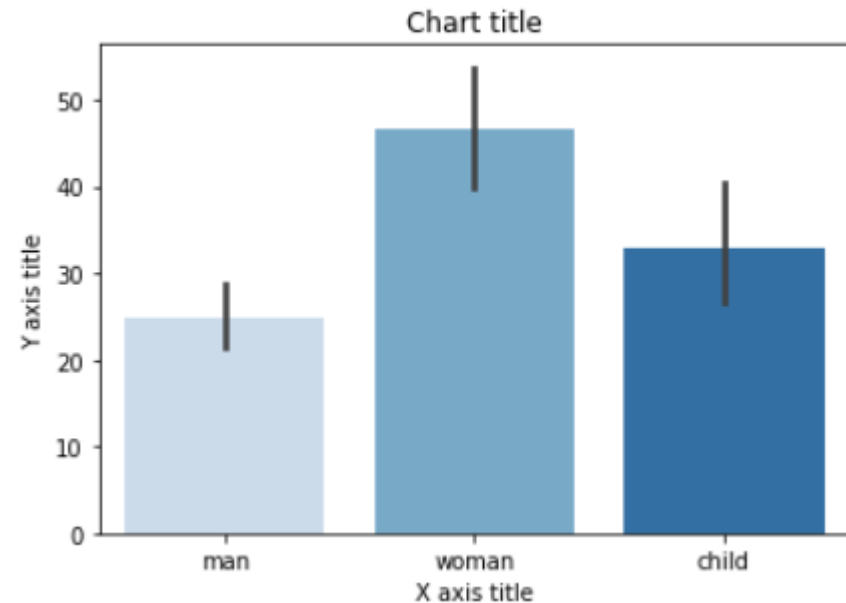
- #Reading the dataset
- `titanic_dataset = sns.load_dataset('titanic')`
- #Creating the bar plot
- `sns.barplot(x = 'fare', y = 'who', data = titanic_dataset, palette = "Blues")`
- #Adding the aesthetics
- `plt.title('Chart title')`
- `plt.xlabel('X axis title')`
- `plt.ylabel('Y axis title')`
- # Show the plot
- `plt.show()`
- `x=""`



Python libraries - Demos

• Seaborn – Column Chart

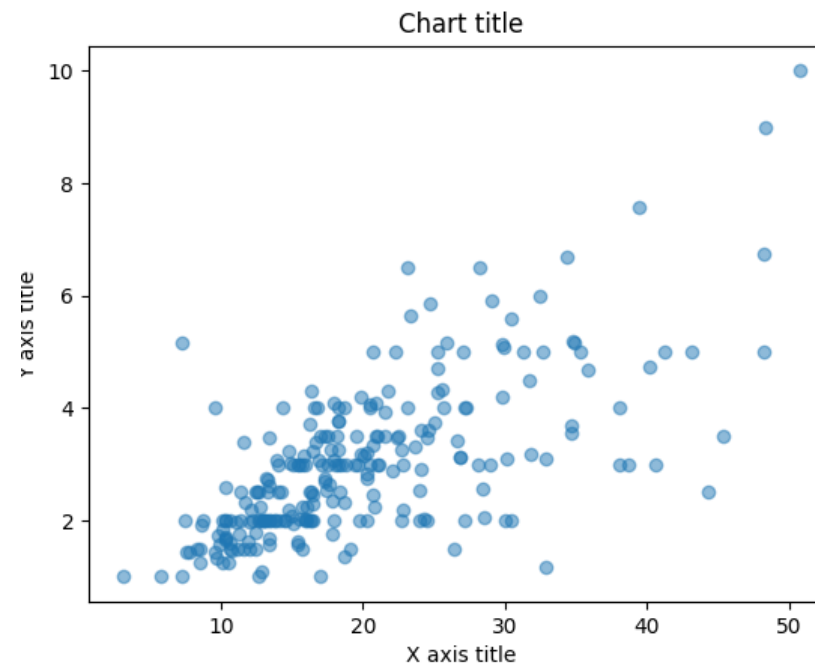
- #Reading the dataset
- titanic_dataset = sns.load_dataset('titanic')
- #Creating column chart
- sns.barplot(x = 'who',y = 'fare',data = titanic_dataset,palette = "Blues")
- #Adding the aesthetics
- plt.title('Chart title')
- plt.xlabel('X axis title')
- plt.ylabel('Y axis title')
- # Show the plot
- plt.show()
- x=""



Python libraries - Demos

- **Matplotlib – Scatter Plot**

- `#Creating the dataset`
- `df = sns.load_dataset("tips")`
- `#Creating the scatter plot`
- `plt.scatter(df['total_bill'],df['tip'],alpha=0.5)`
- `#Adding the aesthetics`
- `plt.title('Chart title')`
- `plt.xlabel('X axis title')`
- `plt.ylabel('Y axis title')`
- `#Show the plot`
- `plt.show()`
- `x=""`

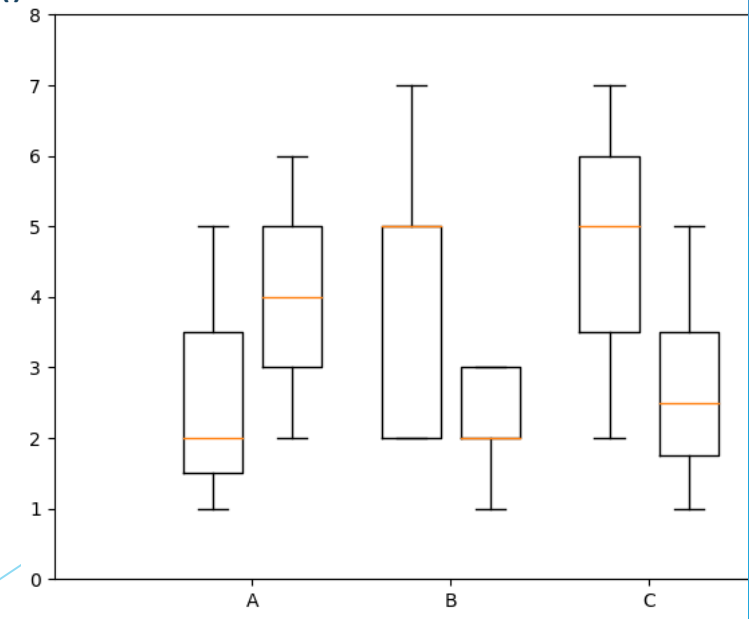


Python libraries - Demos

• Matplotlib – Box Plot

- `from past.builtins import xrange`
- `#Creating the dataset`
- `df_1 = [[1,2,5], [5,7,2,2,5], [7,2,5]]`
- `df_2 = [[6,4,2], [1,2,5,3,2], [2,3,5,1]]`
- `#Creating the box plot`
- `ticks = ['A', 'B', 'C']`
- `plt.figure()`
- `bpl = plt.boxplot(df_1,`
`positions=np.array(xrange(len(df_1)))*2.0-0.4, sym='',`
`widths=0.6)`
- `bpr = plt.boxplot(df_2,`
`positions=np.array(xrange(len(df_2)))*2.0+0.4, sym='',`
`widths=0.6)`
- `plt.plot([], c='#D7191C', label='Label 1')`
- `plt.plot([], c='#2C7BB6', label='Label 2')`

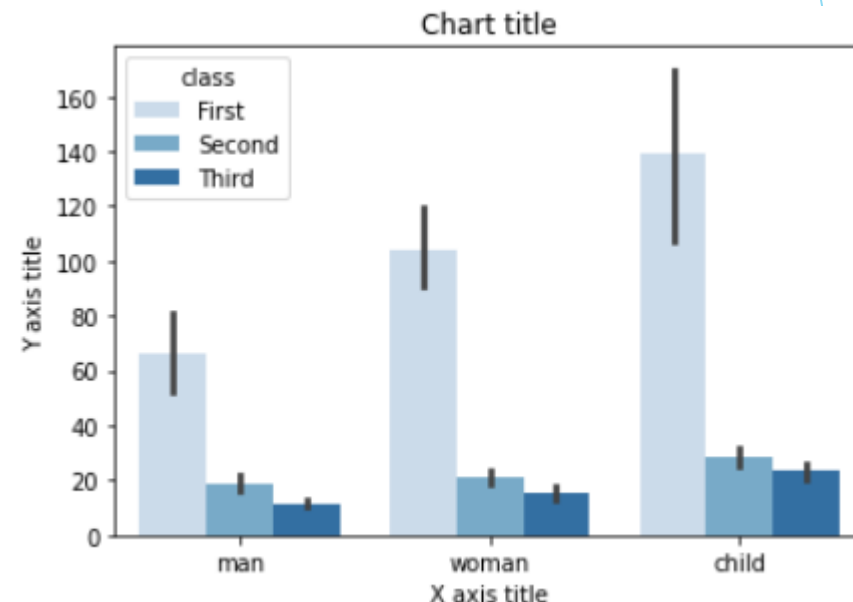
- `#Adding the aesthetics`
- `plt.title('Chart title')`
- `plt.xlabel('X axis title')`
- `plt.ylabel('Y axis title')`
- `plt.legend()`
- `plt.xticks(xrange(0, len(ticks) * 2, 2),`
`ticks)`
- `plt.xlim(-2, len(ticks)*2)`
- `plt.ylim(0, 8)`
- `plt.tight_layout()`
- `#Show the plot`
- `plt.show()`
- `x=""`



Python libraries - Demos

• Seaborn – Grouped Chart

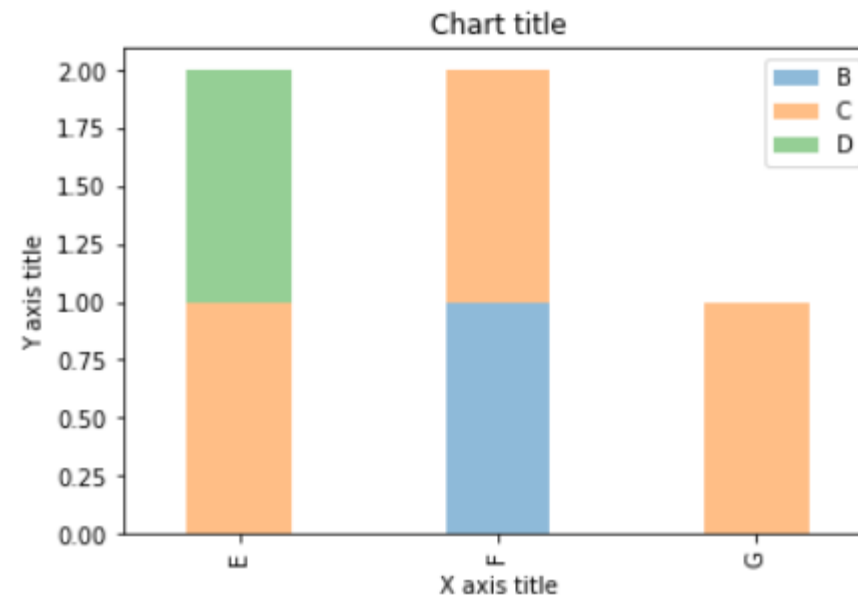
- #Reading the dataset
- titanic_dataset = sns.load_dataset('titanic')
- #Creating the bar plot grouped across classes
- sns.barplot(x = 'who', y = 'fare', hue = 'class', data = titanic_dataset, palette = "Blues")
- #Adding the aesthetics
- plt.title('Chart title')
- plt.xlabel('X axis title')
- plt.ylabel('Y axis title')
- # Show the plot
- plt.show()
- x=""



Python libraries - Demos

• Seaborn – Stacked Bar Chart

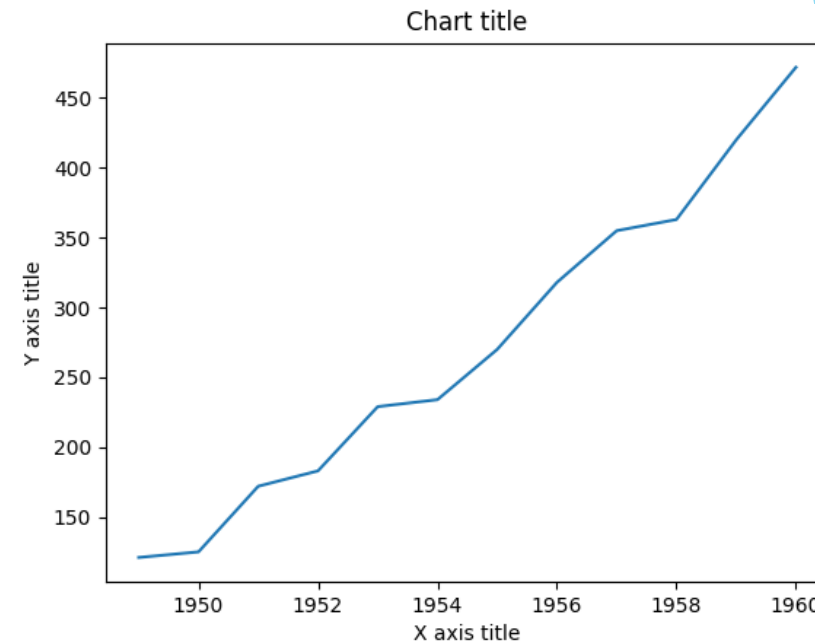
- `#Reading the dataset`
- `titanic_dataset = sns.load_dataset('titanic')`
- `#Creating the stacked bar`
- `dataframe = pd.DataFrame(columns=["A","B", "C","D"], data=[["E",0,1,1], ["F",1,1,0], ["G",0,1,0]])`
- `dataframe.set_index('A').T.plot(kind='bar', stacked=True)`
- `#Adding the aesthetics`
- `plt.title('Chart title')`
- `plt.xlabel('X axis title')`
- `plt.ylabel('Y axis title')`
- `# Show the plot`
- `plt.show()`
- `x=""`



Python libraries - Demos

- **Seaborn – Line Chart**

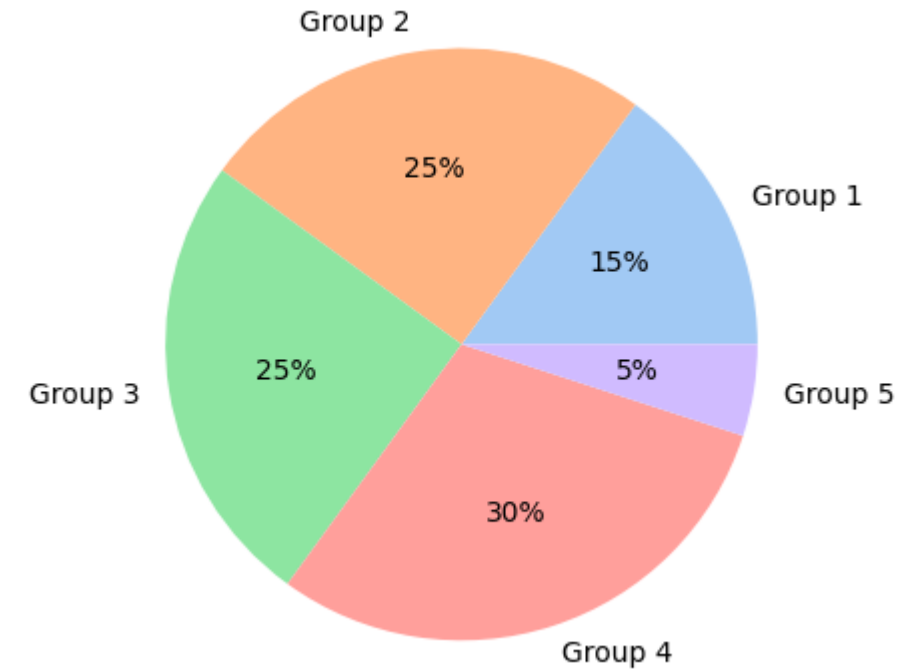
- **#Creating the dataset**
- `flights = sns.load_dataset("flights")`
- `#flights.head()` – do not display
- **#Creating the line chart**
- `may_flights = flights.query("month == 'May'")`
- `sns.lineplot(data=may_flights, x="year", y="passengers")`
- **#Adding the aesthetics**
- `plt.title('Chart title')`
- `plt.xlabel('X axis title')`
- `plt.ylabel('Y axis title')`
- **#Show the plot**
- `plt.show()`
- `x=""`



Python libraries - Demos

• Seaborn – Pie Chart

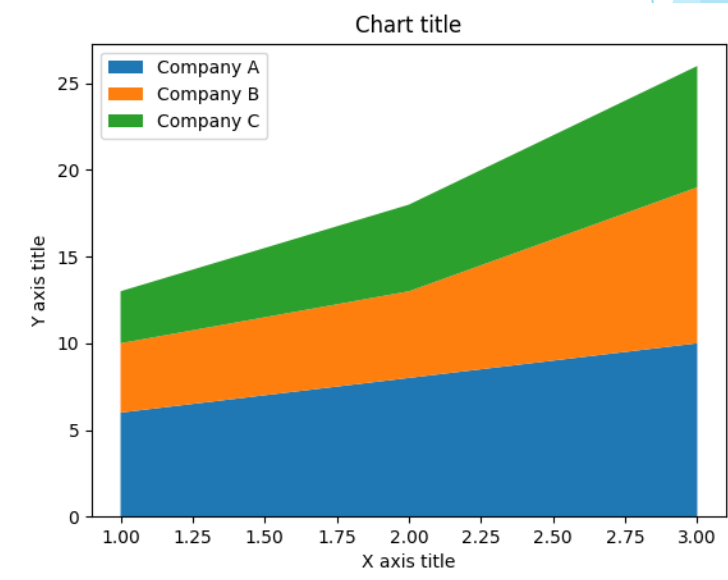
- `import matplotlib.pyplot as plt`
- `import seaborn as sns`
- `#define data`
- `data = [15, 25, 25, 30, 5]`
- `labels = ['Group 1', 'Group 2', 'Group 3', 'Group 4', 'Group 5']`
- `#define Seaborn color palette to use`
- `colors = sns.color_palette('pastel')[0:5]`
- `#create pie chart`
- `plt.pie(data, labels = labels, colors = colors, autopct='%0f%')`
- `#show chart`
- `plt.show()`
- `x=""`



Python libraries - Demos

- **Seaborn – Area Chart**

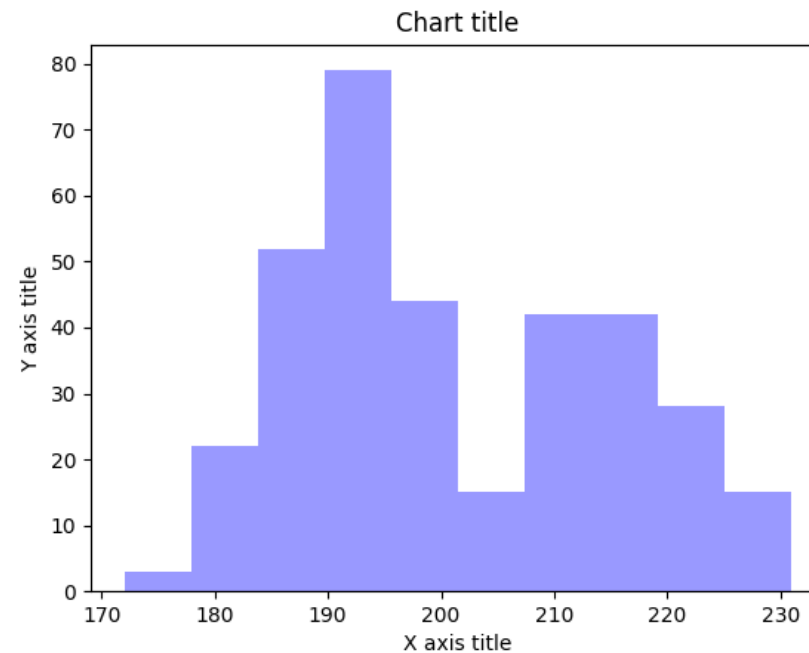
- `# Data`
- `years_of_experience =[1,2,3]`
- `salary=[[6,8,10], [4,5,9], [3,5,7]]`
- `# Plot`
- `plt.stackplot(years_of_experience,salary, labels=['Company A','Company B','Company C'])`
- `plt.legend(loc='upper left')`
- `#Adding the aesthetics`
- `plt.title('Chart title')`
- `plt.xlabel('X axis title')`
- `plt.ylabel('Y axis title')`
- `# Show the plot`
- `plt.show()`
- `x=""`



Python libraries - Demos

- **Seaborn – Column Histogram**

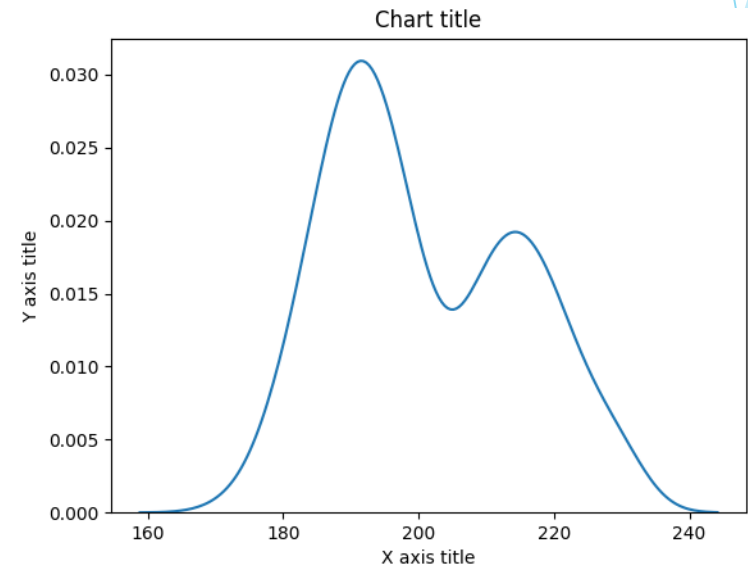
- `#Reading the dataset`
- `penguins_dataframe = sns.load_dataset("penguins")`
- `#Plotting bar histogram`
- `sns.distplot(penguins_dataframe['flipper_length_mm'], kde=False, color='blue', bins=10)`
- `#Adding the aesthetics`
- `plt.title('Chart title')`
- `plt.xlabel('X axis title')`
- `plt.ylabel('Y axis title')`
- `# Show the plot`
- `plt.show()`
- `x=""`



Python libraries - Demos

• Seaborn – Line Histogram

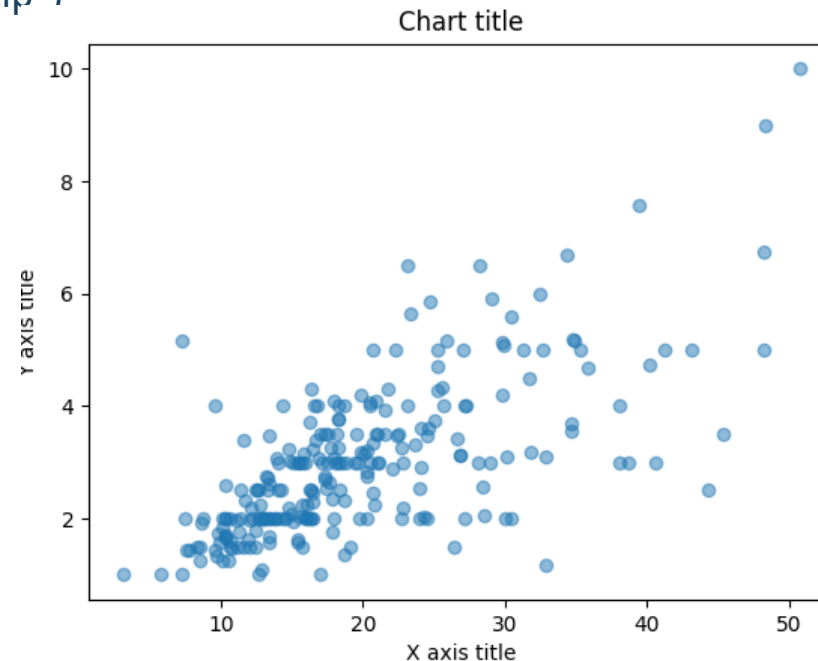
- import matplotlib.pyplot as plt
- import seaborn as sns
- #Reading the dataset
- penguins_dataframe = sns.load_dataset("penguins")
- #Plotting line histogram
- sns.distplot(penguins_dataframe['flipper_length_mm'], hist = False, kde = True, label='Africa')
- #Adding the aesthetics
- plt.title('Chart title')
- plt.xlabel('X axis title')
- plt.ylabel('Y axis title')
- # Show the plot
- plt.show()
- x=""



Python libraries - Demos

- **Seaborn – Scatter Plot**

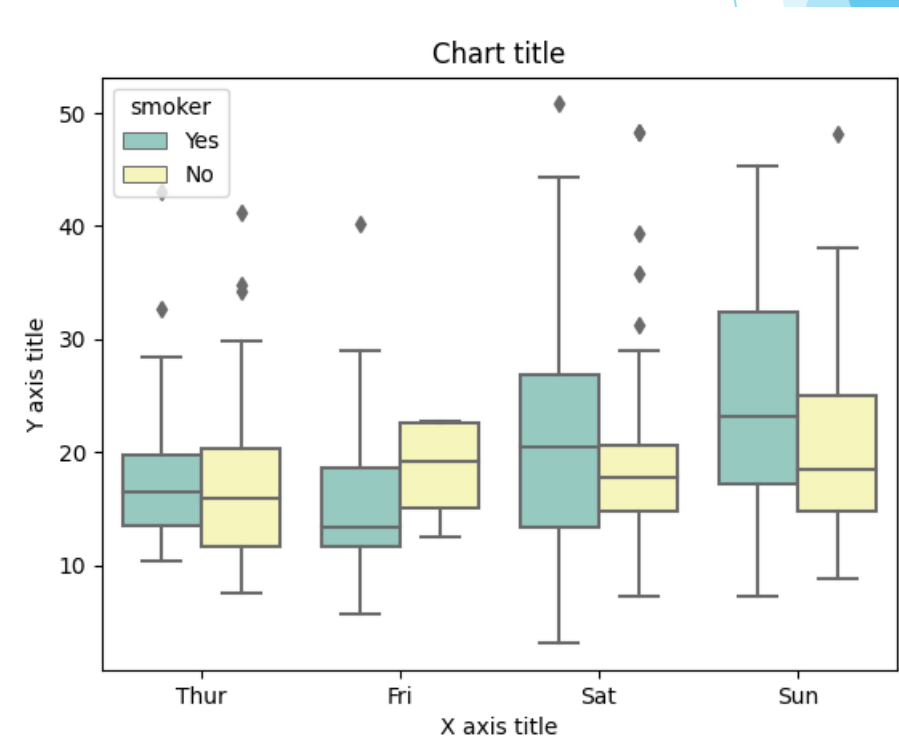
- `#Reading the dataset`
- `bill_dataframe = sns.load_dataset("tips")`
- `#Creating scatter plot`
- `sns.scatterplot(data=bill_dataframe, x="total_bill", y="tip")`
- `#Adding the aesthetics`
- `plt.title('Chart title')`
- `plt.xlabel('X axis title')`
- `plt.ylabel('Y axis title')`
- `# Show the plot`
- `plt.show()`
- `x=""`



Python libraries - Demos

• Seaborn – Box Plot

- `#Reading the dataset`
- `bill_dataframe = sns.load_dataset("tips")`
- `#Creating boxplots`
- `ax = sns.boxplot(x="day", y="total_bill", hue="smoker", data=bill_dataframe, palette="Set3")`
- `#Adding the aesthetics`
- `plt.title('Chart title')`
- `plt.xlabel('X axis title')`
- `plt.ylabel('Y axis title')`
- `# Show the plot`
- `plt.show()`



edureka! www.edureka.co/tableau-certification-trai

CLUSTERING WITH TABPY

Objective: To use an Air BnB dataset and cluster each zipcode based on their hou

<https://www.youtube.com/watch?v=oX1UM87-Oyk>

12:39 / 18:29 How to use TabPy for Clustering? Call us at IN: 9606058406 / US: 18338555775 or visit www.edureka.co

SUBSCRIBE

Recommended Resources

1. Complete Guide to Data Visualization with Python...
<https://towardsdatascience.com/complete-guide-to-data-visualization-with-python-2dd74df12b5e>
2. Tableau and Python / An Introduction - Toan Hoang
<https://tableau.toanhoang.com/tableau-and-python-an-introduction/>
3. Introducing the Python Tableau Data Extract API w...
<https://interworks.com/blog/bbickell/2012/12/06/introducing-python-tableau-data-extract-api-csv-extract-example/>
4. python - Load thousands of CSV files into tableau...
<https://stackoverflow.com/questions/52863882/load-thousands-of-csv-files-into-tableau>
5. csv — CSV File Reading and Writing — Python 3.10....
<https://docs.python.org/3/library/csv.html>
6. Automating a Tableau dashboard with Excel and Pyt...
<https://medium.com/analytics-vidhya/automated-dashboard-with-tableau-excel-python-208805994b7d>
7. From Python to Tableau —Fast and Simple Visualiza...
<https://python.plainenglish.io/from-python-to-tableau-fast-and-simple-visualizations-fe772aa83220>
8. Integrating Python & Tableau. When performing in-...
<https://towardsdatascience.com/integrating-python-tableau-5511dd7102e9>
9. How to pair Tableau and Python for prescriptive a...
<https://www.tableau.com/about/blog/2019/4/leverage-power-tableau-and-python-prescriptive-analytics-104906>
10. Python Pandas – Visualization
https://www.tutorialspoint.com/python_pandas/python_pandas_visualization.htm
11. Plot types — Matplotlib 3.5.0 documentation
https://matplotlib.org/stable/plot_types/index.html
12. Data Visualization in Python | Data Visualization...
<https://www.analyticsvidhya.com/blog/2021/02/an-intuitive-guide-to-visualization-in-python/>
13. Matplotlib: Plot a Function $y=f(x)$
[Plot a Function \$y=f\(x\)\$ in Python \(w/ Matplotlib\) \(scriptverse.academy\)](#)

